FAX COVER SHEET

From: Shakeel Mustafa
24831 Hendon Street
Laguna Hills, CA 92653
Tel: 949-457-1243

Subject:-

Applicant Arguments or Remarks Made in Response to Detailed Office Action

REF: Application/Control Number: 09/848,670

Attention: Ms. Courtney D. Fields

(Primary Patent Examiner)
Art Unit 2137
Fax # 1-703-872-9306
US Patent Office

Number of Pages 33 including this cover page

Date: April 3, 2005

Application/Control Number: 09/848,670 Art Unit: 2137 .

RESPONSES TO THE OBJECTIONS CITED IN DETAILED OFFICE ACTION

DETAILED ACTION

Drawings

Objection

1. The drawings were objected to because reference number "65" is not pointing to"b2" as stated in the specification. Corrected drawing sheets in compliance with 37 CFR . 1.121(d) are required in reply to the Office action to avoid abandonment .....

Response
The attached drawing sheet No. 1 is amended to clearly point out the reference number "65 to "b_2"

Objection:
2. The drawings are objected to as failing to comply with 37 CFR 1.84(p)(5) because they include the following reference character(s) not mentioned in the description: 60, 67, 70, 71., 73, 84, 85, 90, 91, 93, 95, 97, 101, 105, 109, 110, 111, 150, 180, 190, 213, 225, 309, 320, 321, 430, 435, 505, 510, 555, 820, 825, 830, 840, 883, 887, 889. Corrected drawing sheets in compliance with 37 CFR 1.121 (d) or amendment to the specification to add the reference character(s)in the description. In compliance with 37 CFR 1.121(b) are required in reply to the Office action to avoid abandonment of the application.

Response
The attached drawing sheets and/or the specifications are updated and amended to clearly show the reference characters as mentioned above.

Objection:
3. The drawings are objected to as failing to comply with 37 CFR 1.84(p)(5) because they do not include the following reference sign(s) mentioned in the description: 201. Corrected drawing sheets in compliance with 37 CFR 1.121 (d) are required in reply to the Office action to avoid abandonment of the application.

Response
The relevant drawing sheets and/or the specifications are updated and attached to clearly show the reference characters as mentioned above

Objection:
4. The drawings are objected to under 37 CFR 1.83(a). The drawings must show every feature of the invention specified in the claims. Therefore, the step of repeating the encryption rounds "as illustrated in step 221" must be shown or the feature(s) canceled from the claim(s). Also, the existence pf "an inverse function for every function defined as stated in the second paragraph an page 8 regarding fig.3 must be shown or the feature(s) canceled from the claim(s).No new matter should be entered.

Response
The attached drawing sheet shows FIG. 9A which illustrates in detail the step of repeating the encryption rounds. In addition, the explanation related to the FIG. 9A is inserted in the related paragraphs and is underlined.

Objection
5. The drawings are objected to because information segment "S" is not shown in step 215 of fig.9 as stated on page 14 paragraph 2. Also, in fig.4A, it shows M_{min} being greater than M_{max}. Corrected drawing sheets in compliance with 37 CFR 1.121 (d) are required in' reply to the Office action to avoid abandonment of the application

Response:
FIG. 4A has been corrected to properly show that $M_{min} \leq Mn \leq M_{max}$. The "Amended" FIG. 4A showing this change is hereby attached.

Objection
6. . The disclosure is objected to because of the following informalities: Starting with paragraph 2 on page 8 and then pervading throughout the specification, it is not understood whether the second pool contains the inverse functions of the functions within the first pool, or that the second pool contains "another class of plurality of functions" with a "unique inverse function for each of the functions defined in the second pool" as stated in the beginning of the paragraph. If the latter is to be understood as written, then the second pool would appear to not have any relationship to the first as suggested throughout the specification and fig.3.

However, in either case, as stated above, the claimed subject matter of there being a inverse function for every function defined in the pool is not shown in fig.3, which only adds to the confusion.

Response
FIG. 3 has been updated to clearly show how that function's set defined in the first pool is related to the function's set defined in the second pool. In addition, paragraphs [0037] and [0038] have been modified to reflect the changes as presented through FIG. 3.

The specification defines two pools.

The functions defined in the first pool have the following characteristics:

(a) Any types of mathematical or logical functions of arbitrary complexity can be defined in the first pool.

The functions of arbitrary complexity mean that any type of mathematical or logical functions that the participating remote and host processor can handle and process.

(b) The function defined in the first pools are used to encrypt random numbers in such a manner that every operation by the said function(s) enhances random characteristics of the said random number;

(c) An "inverse function" to the functions defined in the first pool may or may not exist.

By definition, the result of a function can be restored to its original value through its inverse function. Mathematically speaking, a function $H(x) = y$ has its inverse function (H^{-1}) if $H^{-1}(y) = x$. If a function does not have an inverse function then it may not be computationally feasible or possible to restore the operated information back to its original form.

The functions defined in the second pool have the following characteristics:

(a) Any types of mathematical or logical functions of arbitrary complexity can be defined in the second pool provided that there must exist an "inverse function" to every function defined in the second pool.

(b) The functions defined in the second pool are used to encrypt data segments in such a manner that every operation by the said function(s) enhances random characteristics of the data segments

(c) Once a function is defined in the second pool, then the corresponding inverse function must also be defined in the second pool.

(d) A defined function in the second pool uniquely points out its peer "inverse function" which is also defined in the second pool.

(e) The defined "inverse functions" are used to decrypt digital data segments which are encrypted by their peer functions in the second pool.

Common examples of functions with inverse functions are the following: The function addition has its inverse function as subtraction; right rotation function with its inverse as left rotation function, etc. The solutions to linear and polynomial equations can also be categorized to have inverse functions only if a predetermined unique solution (single root of the equation) is chosen at both sides.

Objection
On page 9 paragraph 3, it is not understood how Mmin can be greater than Mmax.

Response
The corresponding reference is corrected to show that M_{max} is greater than M_{min}

Objection
On page 10 paragraph 2, fig. 5A shows the Function Bit entries as being "Group #0" and in the specification it states that the entries are "Group #1".Appropriate correction is required.

Response
The corresponding reference is corrected and attached.

Objection
Claim Rejections - 35 *USC* § *101*

7. 35 U.S.C_ 101 reads as follows:

"Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefore, subject to the conditions and requirements of this title".

8. Claims 2-20 rejected under 35 U.S.C. 101 because theses claims are directed to neither a process nor a machine, but rather embraces or overlaps two different statutory classes of invention set forth in 35 U.S.C. 101.

Response
In light of the above discussion, the amended claims are entered as "Method Claims" reflecting appropriate steps to perform a method.

Objections 9-11

***Claim Rejections.* 35 *USC* § 112**

9. The following is a quotation of the second paragraph of 35 U.S.C. 112:

"The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention"

10. Claims 2-20 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. 'Claims that claim both method and system steps of, using an apparatus are indefinite under 35 U.S.C. 112 second paragraph.
11. Claims 1-22 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Claims 1-22 are method claim that contain no method steps. The use 'of the phrase "means for" implies that these are apparatus claims.

Responses 9-11

Applicant respectfully disagrees. As clearly stated in paragraph [0001] of the specification, "The present invention relates to data encryption, particularly to the improvements in processing efficiency of the encryption and decryption of digital information. Furthermore, the present invention relates to encryption involving any type of digital information and to the improvements in processing efficiency of the encryption and decryption of digital information".

As stated earlier, the amended claims are now based on "method claims" clearly showing the method steps. In addition, the claims are amended to reflect the true innovative features and the novelty of the invention.

Objection
12. Claims 1-22 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Claims 1-22 recite numerous limitations such as:

"the first pool", "the second pool", "the numeric value of step (b)", "the digital information segment", "the second function pool as described in step d", "the arbitrary bit segment", "the first pool as described in step e", "the seed arbitrary binary bit segment", "the said arbitrary binary bit segment", "the corresponding inverse function", "said method for operating a digital information processing system that decrypts information", "the encrypted seed binary bit segment", "the first outcome", "the second outcome", and "the seed random number". There is insufficient antecedent basis for this limitation in the

Response
The amended claims now clearly and distinctly point out the subject matter. The scope and the coverage of the phrases as mentioned above are clearly defined. The phrase "the digital information segment" is replaced with "data segment" and so on. Even though, one skilled in art can reasonably argue that the digital information segment means information presented through two level states contrary to analogue information which can theoretically have an infinite number of states. An example of a two level state is a "bit" which is derived from the phrase "binary digit"(bit). The word binary is a composite of "bi" (two) and "nary" (number).

Objection

13. Claims 1-22 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

The terms "any length", "any type", and "any complexity" in claim 1 are relative terms which render the claim indefinite. These terms are not defined by the claim, the specification does not provide a standard for ascertaining the requisite degree, and one of ordinary skill in the art would not be reasonably apprised of the scope of the invention.

The term "large" in claim 5 is a relative term which renders the claim indefinite. The term "large" is not defined by the claim, the specification does not provide a standard for ascertaining the requisite degree, and one of ordinary skill in the art would not be reasonably apprised of the scope of the invention.

The term "any type" in claim 15 is a relative term which renders the claim indefinite. The term "any type" is not defined by the claim, the specification does not provide a standard for ascertaining the requisite degree, and one of ordinary skill in the art would not be reasonably apprised of the scope of the invention.

The term "any information" in claim 19 is a relative term which renders ,the claim indefinite. The term "any information" is not defined by the claim, the specification does not provide a standard for ascertaining the requisite degree, and one of ordinary skill in the art would not be reasonably apprised of the scope of the invention.

The terms "any type", "any arbitrary length segment", and "any information" means" in claim 21 are relative terms which render the claim indefinite. These terms are not defined by the claim, the specification does not provide a standard for ascertaining the requisite degree, and one of ordinary skill in the art would not be reasonably apprised of the scope of the invention

The term "any means mutually agreed" in claim 22 is a relative term which renders the claim indefinite. The term" any means mutually agreed" is not defined by the claim, the specification does not provide a standard for ascertaining the requisite degree, and one of ordinary skill in the art would not be reasonably apprised of the scope of the invention.

The following is a quotation of the first paragraph of 35 U.S.C. 112:

"The specification shall Contain a written description of the invention, and of the manner and process of making and using it, in such full, dear, concise, and exact terms as to enable *any* person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention"

Claim 15 rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the enablement requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to enable one skilled in the art to which it pertains, or with which it is 'most nearly connected, to make and/or use the invention. There are no teachings pertaining "receiving a public key from the host processor" and encrypting system information using "the public key of the host processor". There also is no teachings of decrypting the received information using "the host's private key".

Response

The amended claims fully take into consideration the objections raised. The amended claims are now construed in a way that ensures that the claims coverage will not be indefinite. In addition, amended claims also specify and include the limitations of the related phrases.

Claim Objections

Claim 1 is objected to because of the following informalities: in the limitation "means for defining a plurality of function pool", the word "pool" needs to be plural. In claim 2 element (1), change "functions entries" to "function entries".

Response
The amended claims reflect the suggested changes.

Claim Rejections - 35 use § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

"A person shall be entitled to a patent unless

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States."

Claims 1'-22 are rejected under 35 U.S.C. 102(b) as being anticipated by Johnson et al (US pat 6,052,469).

General Discussion and Response
The following is a thorough discussion which addresses the teachings of Johnson and the related subject matter covered in his invention and compares it with the features and functions covered in my invention.

Under the "Summary of the Invention" Johnson clearly defines the scope of his invention as described under **col. 4 lines 11-20** by stating the following:

"The present invention **contemplates a system for handling key recovery**. It enhances the system described in the copending application of D. B. Johnson et al. **by permitting users to establish the session key using any desired key distribution or key agreement procedure** (e.g., a procedure having the attribute of perfect forward secrecy). The mechanism used to establish the session key is independent of and completely transparent to the cryptographic key recovery procedure. At the same time, the present invention provides a **key distribution procedure** when lacking."

As clearly stated above, Johnson's invention teaches about a key recovery system. It does not disclose or discuss encryption/decryption techniques which are the focal point of my invention. As stated under my "Summary of the Invention" **paragraph [0009]:**

Therefore, **the object of the present invention is to provide a digital information processing method that encrypts information from a plurality of remote processors to a host processor or vice versa**.

This leaves no doubt that Johnson's invention and my invention completely and utterly cover and present **two different field and subject matters in cryptology.**

Furthermore, the following discussion and references illustrate through clear and convincing reasons that the methods and techniques deployed in Johnson's patent and my invention are completely different. As explained by Johnson under **col. 4 lines 21-30:**

The present invention contemplates a new key inversion function that permits the P, Q and R values required by the key recovery procedure to be generated from the secret session key (i.e., by working backwards from the key). That is, the session key is an independent variable and the P, Q and R values are dependent variables. By contrast, in the copending application of D. B. Johnson et al. the P, Q and R values are independent variables and the key is a dependent variable (i.e., the key is derived from the P, Q and R values

Whereas, clearly defined under the working principals of my invention: **paragraph [0009]**

In the presented encryption technique a host and a remote processor, before the start of the encryption procedures, assign and mutually agree upon a certain number of pre-determined bits that are located at pre-determined and specific positions, called Group and Function Bits, of a seed binary bit segment consisting of arbitrary length. The host and the remote also mutually define a first function pool that contains a plurality of mathematical or logical functions of any complexity. Further, the said host and the remote define a second function pool that also contains any type of mathematical or logical functions of any complexity with the condition that there exist a unique inverse mathematical or logical function for each of the functions defined in the second pool.

The methods which can be utilized by the participating host and the remote processors to mutually agree upon the pre-determined positions of the Group and Function Bits in advance are not the focal point of my invention and are left open for further discussion. The concept of mutually agreeing about the positions of the Group and Function Bits in advance is very similar to mutually agreeing on a session key. Symmetric encryption techniques, e.g., DES (Data Encryption Standard), require that the participating host and the remote processors must have the exact same session key in their possession before the start of encryption. Similarly, the encryption/decryption techniques I presented in my invention require that the participating host and remote processors must have advanced knowledge of the positions of Group and Function Bits. How this information is communicated is not the subject matter of my invention. For example, the information about the Group and Function Bits can be exchanged through public/private key infrastructure or over other secure communication links.

Johnson also revealed about using random number called "salt" but in a completely different context

The term "salt" is explained by Johnson in **col. 12 lines 9-19**

FIG. 4 illustrates the procedure 400 used by a sender 102 (FIG. 1) in country X who wishes to send encrypted messages to a receiver 104 in country Y using an independently specified session key. The inputs to the procedure 400 are (1) a secret key; (2) an application-specific

portion of the recovery information; and (3) an optional secret random salt. **A salt is a random value used to increase the randomness of a plaintext. A salt is used only once.** If a secret random salt is provided to the procedure, then it will be called SALT0. Otherwise, SALT0 is derived pseudorandomly from the specified secret key, as described below"

It is a well known and documented fact that in order to help reduce the risk of dictionary attacks, random bytes (so-called "salt") are appended to the original plain text before generating hashes. As cited above, Johnson uses salt (a random number) to increase the randomness of a **plaintext**. In my invention a random number is never directly used or mixed to increase the randomness of plaintext (data segments). The pre-negotiated bits positions within a random number are merely used to identify the functions that are used to encrypt/decrypt data segments. In addition, a random number is not derived pseudorandomly from the specified secret key as per the teaching of Johnson cited above.

Johnson reiterates the fact that his invention is not focused or directed towards encrypting/decrypting data but rather a way to recover a session key. He contends and relies upon the use of these very well known existing encryption techniques as pointed out in **col. 5 lines 62-67** of his patent:

The present invention addresses the communication needs of users and authorized key recovery agents located in different countries. It is applicable to a wide variety of cryptographic algorithms and key lengths. For the purpose of this specification, we will use an example of triple DES with a total key length of 168 bits.

A detailed and step by step comparative analysis is provided in the following sections through the cited references from Johnson's patent. Each reference is cited to address the claim objections raised in the following sections. The amended claims are also being filed to precisely address and cover the novelty and the scope of the invention.

1.0 Claim Objection:

Regarding claim 1, Johnson. teaches a method for operating a digital information processing system that encrypts information from a plurality of remote processors to a host processor or vice versa the method comprising processor executed steps of:

at the host and the remote processors before the start of encryption procedure:

means for assigning and mutually agreeing upon, a pre-determined number of bits that are located at pre-determined and specific positions, called Group and Function Bits, within a seed binary bit segment consisting of any length (col.8 lines 56-61; col1.12 lines 19-30);

1.1A Response to the Objection:

Johnson in Patent number 6,052,469 teaches under paragraphs **col.8 lines 56-61; col.12 lines 19-30:**

Col.8 lines 56-61

The original key K is preprocessed by padding up to 15 zero bits onto the most significant bit positions to form a processed key 302 of n bits, where n is a multiple of 16. The example of FIG. 3A illustrates the case of a sender, in country X, who makes use of an r-bit R value and a receiver, in country Y, who uses an R of 0 bits.

In the above description produced by Johnson, there is no disclosure or any reference of "**assigning and mutually agreeing upon, a pre-determined number of bits that are located at pre-determined and specific positions, called Group and Function Bits, within a seed binary bit segment consisting of any length.**" A random number, as pointed out in the specifications of my invention, can consist of a binary bit segment. Johnson does not talk about or refer to a random number where specific bit locations within the random number are selected in a pre-negotiated manner and based on the numeric values of these identified bits, consecutively selecting plurality of functions to encrypt data segments of arbitrary length. The method disclosed in my invention does not dictate or impose any restrictions on the length of a data segment. A data segment can consist of an arbitrary length as long as the system recourses utilized by the sender and the receiver devices can handle and process the segment.

In the above reference Johnson talks about padding the original key "K" and this teaching is not discussed through any reference of my invention.

1.1B. Response to the Objection:
Johnson in Patent number 6,052,469 teaches under paragraphs **col.12 lines 19-30:**

Col.12 lines 19-30
Referring to FIG. 4, the sender 102 and receiver 104 first establish a secret session key K (step 402). The sender 102 and receiver 104 may use whatever key distribution or key agreement procedure that they desire for establishing the key K. Typically the session key K will be a symmetric encryption key that is used both by the sender 102 to encrypt messages for the receiver 104 and by the receiver to decrypt messages from the sender. If public keys are used as session keys, then the procedure may be used by defining K=(PKa,SKa), where PKa is the public encryption key used by the sender 102 and SKa is the corresponding secret decryption key used by the receiver 104

The above reference does not discuss or disclose any method about **"assigning and mutually agreeing upon, a pre-determined number of bits that are located at pre-determined and specific positions, called Group and Function Bits, within a seed binary bit segment consisting of any length."** The above cited reference suggests establishing a "secret session key" between a sender and receiver. My invention does not disclose or depend on in any way or form on establishing a "secret session key" between sender and receiver. As a matter of fact, in my invention there is no need to establish a "secret session key" between sender and receiver. There is a fundamental difference between establishing a "secret session key" between a sender and receiver and "selecting bits from the pre-negotiated locations within a **random number**". This described concept possesses a central and pivotal role in the innovative subject matter detailing my invention. Johnson further reveals that the "session key K will be a symmetric encryption key" which is used to encrypt and decrypt messages. My invention is not about recovering or establishing a session key.

The following are a few other very crystal clear and primary differences between Johnson's teachings and my invention.

Johnson's teachings
Both the sender and receiver must communicate with each other through some covert means so they can exchange the "secret session key." If an eavesdropper happens to capture the transmission of the "secret session key," the encryption process based on the session key will no longer be secured.

According to the Data Encryption Standard (DES) **two keys are symmetric** provided that both the keys

have the **exact same length** and have the **same bits in all the bit locations** within the given binary segment.

My invention
As stated earlier there is no concept of "session key" in my invention. The sender and the receiver mutually agree on the **locations of pre-determined number of bits located within a random number of an arbitrary length** through a set of pre-negotiated rules. There is no session key involved

The length of the random number can be completely arbitrary and may change every time a sender transmits a random number to the receiver. The locations of the Group and Function Bits assigned within a random number can also be optionally changed with length. This provides another level of protection against an eavesdropper who is trying to determine the locations of the Group and Function Bits within a variable size of a random number.

It is not required to have covert communication to exchange a random number. In the event an eavesdropper intercepts the transmitted random number he cannot tell which bits locations (Group and Function Bits) are being used to identify the functions chosen for encryption between the sender and receiver as disclosed in the details of my invention.

1.2. Claim Objection:
The following part of the Claim # 1 is rejected based on the citation reference quoted from Johnson **col.5lines 62-67; col.18 lines 48-62**

"means for defining a plurality of function pool containing any type of mathematical or logical functions of any complexity";

1.2A Response to the cited Objection
In **col.5 lines 62-67** Johnson teaches:

col.5 lines 62-67

The present invention addresses the communication needs of users and authorized key recovery agents located in different countries. It is applicable to a wide variety of cryptographic algorithms and key lengths. For the purpose of this specification, we will use an example of triple DES with a total key length of 168 bits.
As clearly stated in the above reference there is no disclosure or reference about "**defining a plurality of function pools containing any type of mathematical or logical functions of any complexity.**" Johnson talks about the scope of his invention as a "key recovery" mechanism by the "authorized key recovery agents located in different countries." The type of "key length" he is referring to is the length of "symmetric session key" as stated in Col.12 lines 19-30 quoted above. He reiterates this concept by specifically quoting "triple DES with a total key length of 168 bits."

1.2B Response to the cited Objection
Further, in **col.18 lines 48-62** Johnson teaches:

Col.18 lines 48-62
Referring to FIG. 12, the information required by the key recovery system of the present invention is stored in a table 1200 called the global communications policy table. Table 1200 is for illustrative purposes only. In an actual implementation the data would be stored

appropriately, perhaps in separate tables, one specifying the key recovery agents' public keys and one specifying the rules. Table 1200 contains information allowing the system to calculate the sizes of the keys and P, Q and R for specific algorithms and users located in different countries. It may also contain the public keys of key recovery agents authorized for each country. The numbers in table 1200 are examples only to demonstrate the kind of flexibility the present invention permits. The variations are virtually unlimited. In particular, each country may have many key recovery agents.

As clearly stated above by Johnson there is no disclosure or reference about defining a "function pool" that contains "mathematical or logical functions" to be used for encryption/decryption. It talks about storing the data in separate tables which is not even a consideration in my invention. The scope of his invention is a key recovery system whereas the "information required by the key recovery system" is stored in a policy table. In a nutshell, it talks about recovering the keys "P, Q and R" and does not disclose any technique in regards to data encryption which is the scope of my invention.

1.3 Claim Objection:
The following part of the Claim # 1 is rejected based on the citation reference quoted from Johnson **col.13 lines 17-30:**

"means for establishing a unique relationship between the functions defined in the first pool with the functions defined in the second pool sequentially identical at both the host and the remote processors (col.13 lines 17-30)"

1.3A Response to the cited Objection
In **col.13 lines 17-30** Johnson teaches:

Col. 13 lines 17-30
Following signature validation, the receiver 104 validates the received encrypted P and Q values 602-608 by duplicating the steps performed by the sender 102. Thus, the receiver 104 generates P, Q and R values from the secret session key K (step 506), generates salt values from the key K (step 508) and generates encrypted P and Q values from the salt values (step 510), with steps 506-510 being identical to the steps 404-408 performed by the sender 102. The receiver 104 then compares the encrypted P and Q values thus generated for equality with the set of encrypted P and Q values 602-608 received from the sender 102; the receiver also checks the recovery information 610 received from the sender for consistency with the similar information maintained by the receiver (step 1108).

The above citation does not disclose any reference for the use of plurality of functions defined in a pool. It talks about ways how the receiver generates P, Q and R values from the secret session key. Again, the scope of this paragraph is how to generate "P, Q and R values" from the secret session key K. It does not address any of the techniques involving data encryption through the random numbers which is a focal point of my invention. In addition, the related claim is hereby amended to incorporate the changes in light of the above reference and discussion.

1.3B Response to the cited Objection
The response to the next reference as cited under **col.18 lines 48-62** is discussed under **1.2B Response to**

the Objection. As stated earlier, the cited reference does not discuss defining functions in the first pool with the functions defined in the second pool. Its focus is how to recover a session key. It talks about "public keys of key recovery agents authorized for each country" and does not address any technique that is required to carry out an encryption/decryption procedure of a data segment.

1.4 Claim Objection:
The following part of the Claim # 1 was rejected based on the citation reference quoted from Johnson **col.17 lines 1-11)**

"means for defining a number 'N' which indicates the total number of rounds used for encryption/decryption process (col.17 lines 1-11)".

1.4A Response to the cited Objection
In **col.17 lines 1-11** Johnson teaches:

Col. 17, lines 1-11
Referring to FIG. 11, H(T1), the P or Q value, the indicator, and the salt are formatted in a block (1102), padded as necessary on the left with 0 bits, and encrypted with the public key of the key recovery agent, preferably using the enhanced optimal asymmetric encryption (EOAE) procedure described in the above-identified article of D. B. Johnson et al. As described in that article, the EOAE procedure contemplates first subjecting the formatted block 1102 to a plurality of masking rounds 1104 (in which one input half is alternating used to mask the other input half) before encrypting (1106) the result of the masking rounds.

In the above reference there is no indication about mutually agreeing upon encryption/decryption rounds being performed on data segments. It talks about "masking rounds" in reference to recovering session keys which are completely different from actually performing variable rounds of encryption/decryption rounds on data segments. In my disclosure the **number 'N' which indicates the total number of rounds used for the encryption/decryption process** is variable and not fixed. This number is dynamically determined through the information contained in a random number or through the encrypted version of the random number. Johnson defines the masking round operation as "in which one input half is alternating used to mask the other input half." This concept is completely different from the definition of an encryption round according to the techniques disclosed in my invention. In addition, the related claim is herby amended to incorporate the changes in light of the above reference and discussion.

1.5 Claim Objection:
The following part of the Claim # 1 was rejected based on the citation reference quoted from Johnson **col.13 lines 19-24**

"(a) means for generating and sending a seed arbitrary binary bit segment consisted of any length to the host processor (col.13 lines 19-24);"

1.5A Response to the cited Objection

In **col.13 lines 19-24** Johnson teaches:

Col.13 lines 19-24

Thus, the receiver 104 generates P, Q and R values from the secret session key K (step 506), generates salt values from the key K (step 508) and generates encrypted P and Q values from the salt values (step 510), with steps 506-510 being identical to the steps 404-408 performed by the sender 102

The above citation has no reference of "**generating and sending a seed arbitrary binary bit segment that consists of any length to the host processor.**" The scope of Johnson's invention is to recover a session key. It does that by generating P, Q and R values from the secret session key K. It does not generate and exchange a random number which can consist of any **arbitrary binary bit segment** between a host and remote processor to be used for the encryption and decryption of data segments. In addition, the related claim is herby amended to incorporate the changes in light of the above reference and discussion.

1.6 Claim Objection:
The following part of the Claim # 1 was rejected based on the citation reference quoted from Johnson **col.13 lines 17-30**

"(b) means for processing the seed arbitrary binary bit segment at the remote processor (col.13 lines 17-30)"

1.6A Response to the cited Objection
The response to the cited reference is discussed under **1.3A Response to the Objection.** The processing of the random number in my invention means utilizing the pre-negotiated bit locations called Group and Function Bits in a random number which in turn select a range of variable functions. The identified functions are used to encrypt/decrypt data segments. Johnson presents a completely different implementation and concept which refers to a way that "validates the received encrypted P and Q values."

1.7 Claim Objection:
The following part of the Claim # 1 was rejected based on the citation reference quoted from Johnson **col.13 lines 17-30**

"(c) means for producing a numeric number value based on the bit values of the Group and Function Bits as defined in the said arbitrary binary bit segment (col.13 lines 17-30)"

1.7A Response to the cited Objection
The response to the cited reference is discussed under **1.3A Response to the Objection.** There is absolutely no indication of processing the pre-determined location of bits in a random number called Group and Function Bits in the above cited reference. As stated earlier, the cited reference deals with producing a session key K from P, Q and R values and not the procedures presented in my invention for encrypting/decrypting data segments.

1.8 Claim Objection:
The following part of the Claim # 1 was rejected based on the citation reference quoted from Johnson **col.14 lines 42-55**

"(d) means for selecting a single or plurality of mathematical or logical functions from the first pool based upon the numeric number value of step (b) (col.14 lines 42-55)"
1.8A Response to the cited Objection

In **col.14 lines 42-55** Johnson teaches:

Col. 14 lines 42-55
The sender's and receiver's first and second key recovery agent key headers 914, 918, 924, 928 **contain information on the public keys belonging to the key recovery agents**. Referring to FIG. 10, each key header contains an encryption algorithm ID 946 (specifying the public key algorithm to be used), the key length 948 in bits of the public key, and a key ID 950. The key ID 950 permits the receiver to determine the public keys under which the P and Q values are encrypted. The receiver needs these public keys in order to validate the encrypted P and Q values 602 and 604. The key IDs 950 permit the key recovery agents to determine the public keys under which the P and Q values are encrypted, and hence the private keys needed to decrypt the P and Q values.

The above citation presents no reference **for selecting a single or plurality of mathematical or logical functions from the first pool based upon the numeric number value of Group or Function Bits.** Johnson talks about recovering session keys through the use of public keys. It also refers to the public key under which P and Q values are encrypted. It also has a reference that each key header contains an encryption algorithm ID 946. In my invention there are no key headers to contain encryption algorithms.

1.9 Claim Objection:
The following part of the Claim # 1 was rejected based on the citation reference quoted from Johnson **col.14 lines 45-47:**

"(e) means for identifying the corresponding single or plurality functions from the second pool (col.14 lines 45-47)"

1.9A. Response to the cited Objection
In **col.14 lines 45-47** Johnson teaches:

Col.14 lines 45-47
The sender's and receiver's first and second key recovery agent key headers 914, 918, 924, 928 contain information on the public keys belonging to the key recovery agents. Referring to FIG. 10, each key header contains an encryption algorithm ID 946 (specifying the public key algorithm to be used), the key length 948 in bits of the public key, and a key ID 950

There is no indication that Johnson teaches to identify "single or plurality of functions from the second pool." Johnson even does not teach that there would be two function pools; one for encrypting a random number only and the other one for encrypting/decrypting data segments. The theme of Johnson's teaching is how to recover session keys for key recovery agents and not encrypting/decrypting data segments. Again, he teaches about the scenario in which each key header contains an encryption algorithm ID 946 (specifying the public key algorithm to be used). In my case there are no public keys or key headers containing encryption algorithm IDs.

1.10 Claim Objection:
The following part of the Claim # 1 was rejected based on the citation reference quoted from Johnson

col.18 lines 54-62:

"(g) means for encrypting the digital information segment through operating single or plurality of mathematical or logical functions selected from the second function pool as described in step d (col.18 lines 54-62);"

1.10A Response to the cited Objection
In **col.14 lines 45-47** Johnson teaches:

Col. 18 lines 54-62
Table 1200 is for illustrative purposes only. In an actual implementation the data would be stored appropriately, perhaps in separate tables, one specifying the key recovery agents' public keys and one specifying the rules. Table 1200 contains information allowing the system to calculate the sizes of the keys and P, Q and R for specific algorithms and users located in different countries. It may also contain the public keys of key recovery agents authorized for each country. The numbers in table 1200 are examples only to demonstrate the kind of flexibility the present invention permits. The variations are virtually unlimited. In particular, each country may have many key recovery agents.

The above reference does disclose or suggest **encrypting the digital information segment through operating single or plurality of mathematical or logical functions selected from the second function pool.** In the above citation Johnson talks about building a table that can store data. In my invention, there is no indication to store data in "separate tables." It talks about calculating the sizes of the keys and P, Q, R for the specific algorithm. In my invention there is no calculating the sizes of keys for a specific algorithm. The functions chosen to encrypt/decrypt data depend upon the outcome of the pre-negotiated bits found in the random number and can be different each time a data segment is encrypted.

1.11 Claim Objection:
The following part of the Claim # 1 was rejected based on the citation reference quoted from Johnson **col.5 lines 62-67**

"means for encrypting the arbitrary binary bit segment through operating single or plurality of mathematical or logical functions selected from the first pool as described in step c (col.5lines 62-67);"

1.11A Response to the cited Objection
In **col.5 lines 62-67** Johnson teaches:

col.5 lines 62-67
The present invention addresses the communication needs of users and authorized key recovery agents located in different countries. It is applicable to a wide variety of cryptographic algorithms and key lengths. For the purpose of this specification, we will use an example of triple DES with a total key length of 168 bits.

As clearly stated above Johnson does not mention "**encrypting the arbitrary binary bit segment through operating single or plurality of mathematical or logical functions selected from the first pool.**" As discussed earlier, Johnson talks about "session key" in reference of implementing the

encryption algorithms which are based on symmetric encryption, for example, triple DES or other private/public key infrastructure.

1.12 Claim Objection:
The following part of the Claim # 1 was rejected based on the citation reference quoted from Johnson **col.5 lines 62-67**

"means for replacing the seed arbitrary binary bit segment with the encrypted arbitrary binary bit segment and using it as a new seed arbitrary binary bit segment (col.16 lines 46-58; col.17 lines 47-50);"

1.12 A Response to the cited Objection
In **col.16 lines 46-58** Johnson teaches:

Col. 16 lines 46-58
The salts (SALT<for Px>, SALT<for Qx>, SALT<for Py>, and SALT<for Qy>) protect the encrypted P and Q values 602-608. The salt values (SALT<for Px>, SALT<for Qx>, SALT<for Py>, and SALT<for Qy>) are specifically constructed to be different. In the case where SALT0 is a secret random value specified as an input to the encryption procedure, this guarantees that every block to be encrypted for a key recovery agent has a dependency on SALT0 (a secret random 160-bit value independent of the key). In the case where SALT0 is pseudorandomly generated from the key, this guarantees that every block to be encrypted for a key recovery agent has a dependency on the entire key in a pseudorandom way.

The term "salt" is explained by Johnson in **col. 12 lines 9-19**

FIG. 4 illustrates the procedure 400 used by a sender 102 (FIG. 1) in country X who wishes to send encrypted messages to a receiver 104 in country Y using an independently specified session key. The inputs to the procedure 400 are (1) a secret key; (2) an application-specific portion of the recovery information; and (3) an optional secret random salt. **A salt is a random value used to increase the randomness of a plaintext. A salt is used only once.** If a secret random salt is provided to the procedure, then it will be called SALT0. Otherwise, SALT0 is derived pseudorandomly from the specified secret key, as described below"

In the above cited reference Johnson does not disclose or teach about "**replacing the seed arbitrary binary bit segment with the encrypted arbitrary binary bit segment and using it as a new seed arbitrary binary bit segment.**" It is a well known and documented fact that in order to help reduce the risk of dictionary attacks, random bytes (so-called "salt") are appended to the original plain text before generating hashes. As cited above, Johnson uses salt (a random number) to increase the randomness of a **plaintext**. In my invention a random number is never directly used or mixed to increase the randomness of plaintext (data segments). The pre-negotiated bits positions within a random number are merely used to identify the functions that are used to encrypt/decrypt data segments. In addition, a random number is not derived pseudorandomly from the specified secret key as per the teaching of Johnson cited above.

1.12B Response to the cited Objection

In **col.17 lines 47-50)** Johnson teaches:

If there are "i" such blocks, then "i" different salt values are calculated, thereby ensuring that a different (but predictable) salt value is calculated for each of the m-bit blocks to be encrypted.

In order to fully understand the meaning and coverage of the above paragraph we need to read the entire context from **col. 17 lines 42-50**

When the length of P or Q is >m (**where m is the maximum length of P or Q that can be encrypted with the public key of the intended key recovery agent**), the value (P or Q) is divided into blocks of m bits. The last block may be a short block. If there are "i" such blocks, then "i" different salt values are calculated, thereby ensuring that a different (but predictable) salt value is calculated for each of the m-bit blocks to be encrypted. For example, suppose that m=256 and the length of P is 512 bits. In that case, P is split into two blocks of size 256 bits and two salt values are calculated using the above described algorithm

Again in the above cited reference Johnson does not disclose or teach about "**replacing the seed arbitrary binary bit segment with the encrypted arbitrary binary bit segment and using it as a new seed arbitrary binary bit segment.**" The above reference presents a situation where P or Q is greater than 'm'. In this situation Johnson recommends to divide the value P and Q into blocks of 'm' bits.

As mention earlier, my invention does not teach or depend on the teachings or use of so called 'P' and 'Q' which are strictly used in Johnson's patent to recover the session key 'K'. Johnson divides the value 'P' or 'Q' into blocks of 'm' bits. In my invention there is absolutely no requirement to either divide the data segment or a random number. As a matter of fact, either a data segment or a random number can consist of any number of bits and still the techniques disclosed in my invention can encrypt/decrypt the information. The salt value (random number value) are calculated for each of the 'i' blocks. In my invention, a random number is actually encrypted through the use of functions defined in the first pool. In my invention, there is no correlation between 'i' such blocks with 'i' different salt values (random number) for the reason that my invention does not teach about 'P', 'Q' or block of 'm' bits. The above reference presents a situation that P and Q can be encrypted by the public key.

1.13 Claim Objection:
The following part of the Claim # 1 was rejected based on the citation reference quoted from Johnson **col.5 lines 62-67**

"means for repeating the steps (b) to (h) N' times and then transmitting the resulting encrypted digital information segment to the said host (col.17 lines 1-11);"

1.13A Response to the cited Objection
In **col.17 lines 1-11** Johnson teaches:

Col. 17, lines 1-11
Referring to FIG. 11, H(T1), the P or Q value, the indicator, and the salt are formatted in a block (1102), padded as necessary on the left with 0 bits, and encrypted with the public key of the key recovery agent, preferably using the enhanced optimal asymmetric encryption

(EOAE) procedure described in the above-identified article of D. B. Johnson et al. As described in that article, the EOAE procedure contemplates first subjecting the formatted block 1102 to a plurality of masking rounds 1104 (in which one input half is alternating used to mask the other input half) before encrypting (1106) the result of the masking rounds.

In the above description there is no indication about repeating the encryption/decryption rounds in accordance with the disclosed techniques described in my invention. As mentioned before, the cited reference talks about plurality of masking rounds in reference to extracting session keys. As defined further, the nature of the masking round suggested by Johnson constitutes using "one input half is alternating used to mask the other input half." In my invention, plurality of mathematical or logical functions operate as a round of encryption on the data segments and not on the session keys.

Response and General Discussion to Objections on Claim No. 21

The majority of the references cited for Objections under Claim No. 21 have already been specifically addressed under Responses to Objections on Claim No. 1. The following discussion is divided into two sections. The first section (**Section 1**) makes a comparison between the subject area of Johnson teachings and my invention. The second section (**Section 2**) discusses the working principals and techniques used in Johnson's patent versus the techniques used in my invention.

Section 1

1.0 Johnson's Patent

Under "Field of the Invention," col.1 lines 11-15, Johnson reveals the scope of his invention as

"This invention relates to a cryptographic key recovery system and, more particularly, to a key recovery system that is interoperable with existing systems for establishing keys between communicating parties"

In the entire discussion that follows he concentrates on a key recovery system that can be used by law enforcement agencies to recover "session keys" used in symmetric encryption systems such as Data Encryption System (DES). In order to recover the "session key" Johnson proposes under **col. 4 lines 21-30**

The present invention contemplates a new key inversion function that permits the P, Q and R values required by the key recovery procedure to be generated from the secret session key (i.e., by working backwards from the key). That is, the session key is an independent variable and the P, Q and R values are dependent variables. By contrast, in the copending application of D. B. Johnson et al. the P, Q and R values are independent variables and the key is a dependent variable (i.e., the key is derived from the P, Q and R values).

He further teaches under **col.5 lines 4-12**

"The present invention contemplates sending a session context which contains enough information: (1) to allow the parties to agree on

keys, or to employ any independently established keys; (2) to allow the receiver to verify the associated key recovery information; (3) to allow authorized entities the ability to recover components of the keys; and (4) to allow the authorized entities to verify the correctness of the key recovery information given by the key recovery agents."

Johnson further states that the scope of his invention is not to present a technique for encrypting/decrypting information. He contends on using his "key recovery" technique in conjunction with the existing and standard data encryption techniques. As Johnson states under col. 5 lines 29-49

The advantages of the present invention may be briefly summarized. It is an add-on solution interoperable with any key distribution procedure, and it provides for key distribution when lacking. It allows recovery of lost cryptographic keys by recovery agents. No user keys are held by key recovery agents. Key recovery agents have no role in generating user keys. It is a multi-way key recovery scheme. It enables strong cryptography world-wide. It addresses the needs of legitimate and authorized law-enforcement, while at the same time addressing inherent weaknesses in other key recovery proposals (e.g., the requirement of an infrastructure to establish a user key or the requirement of a special hardware device). It has no limit on key lengths, nor any algorithm restrictions. It is interoperable with all key exchange mechanisms. Only session encryption keys are recoverable. Key recovery information is made partially available in accordance with policy. A uniform (encryption algorithm-independent) work factor is provided for full key recovery. Since the present invention uses only encryption, it would be hard to subvert the scheme to use it for a bulk data confidentiality channel.

1.1. Applicant's Patent Application

As clearly stated under the "Abstract" section of my Application:

"The present invention provides a simple but extremely robust encryption method and system for encrypting any type of digital information that consists of any arbitrary length. A host can simultaneously maintain plurality of encrypted communication sessions with several remotes"

Note: Thee terms like "any type" or "any length" are amended and clarified in the related specifications/claims.

I further disclose under the "Background of the Invention"[0001] section of my Application:

"The present invention relates to data encryption, and more particularly to the improvements in processing the efficiency of the encryption and decryption of digital information. Furthermore, the present invention relates to encryption involving any type of digital information, and to the improvements in processing efficiency of the encryption and decryption of digital information. The major problem

that exists with current encryption methods is that of speed. As the level of encryption complexity increases, the processing speed requirements also increase by many folds."

As clearly shown throughout the above discussion, these two inventions address two completely different areas. The two inventions cover and present two different subject fields in cryptography: one presents the techniques of recovering and managing keys (Johnson's invention) and the other (my invention) presents a method that can be used for encrypting/decrypting data information segments.

Section 2

2.0 Johnson's Patent

Johnson teaches about recovering the session key "K" through the use of inversion functions. Johnson further explains about this procedure in **col. 8 lines 42-49** as follows:

"FIG. 3A shows an exemplary key inversion function 300 for generating key recovery values P, Q and R from a session key K or vice versa. While the exemplary key inversion function 300 generates P and Q values for two key recovery agents per user, the key inversion function can produce any number of outputs to handle the case where the sender and receiver each have more than two key recovery agents or only a single key recovery agent".

Clearly, from the above discussion, Johnson's teachings are primarily focused on processing key recovering values either from P, Q and R to generate K or vice versa..

He further discloses the inputs required for the key inversion functions in **col. 8 lines 50-55** as follows:

"Key inversion function 300 requires the following inputs: (1) the key K; (2) the length of the key K in bits; (3) the length of R in bits, denoted r; and (4) recovery information 610 (FIG. 9) to be described, including the first and second key recovery agent IDs 912, 916, 922, 926 for the communicating parties 102, 104"

In the above description he has laid out the steps that are required to execute the "Key inversion function" which is the primarily backbone focus in his patent. Not even a single step described above has any resemblance to the methods described in my invention for data encryption.

2.1 Applicant's Patent Application

In my invention, there is no procedure to recover a key through the inversion functions. In my invention, a pre-determined number of bits'locations are mutually agreed upon in advance in a random number between a sender and a receiver are used to calculate a numeric value. Based upon the results, different functions sets defined in a function pool are selected to encrypt/decrypt data segments.

As presented in the above discussion there is no correlation between the subject matter (Key Recovery system) and the techniques presented by Johnson with the subject matter (Encrypting/Decrypting data segments through random number) and techniques used in my invention.

Responses to the individual Objections as cited in reference to Claim No. 21

21.1 Claim Objection:

The following part of the Claim # 21 was rejected based on the citation reference quoted from Johnson

(col.12 lines 39-49)

"Regarding claim 21, Johnson teaches a method for operating a digital information processing system that encrypts information from a plurality of transmitting devices to a receiving device or vice versa the method comprising processor executed means for using the information contained in the random number to identify single or plurality of unique mathematical or logical functions identical at the both transmitting and the receiving devices (col.12 lines 39-49);"

21.1A Response to the cited Objection
In **(col.12 lines 39-49)** Johnson teaches:

"The sender 102 then generates several salt values, using the secret session key K, in a manner to be described below (step 406).

Referring to FIG. 6, the sender 102 uses the generated salt values, together with the public keys of the key recovery agents 108-114 and other information, to generate encrypted P and Q values 602, 604 for country X and encrypted P and Q values 606, 608 for country Y (step 408). Using this information, the sender 102 creates a session context 612 (FIG. 6) consisting of the concatenation of encrypted P and Q values 602-608 and recovery information 610 (step 410)."

The term "salt" is explained by Johnson in **col. 12 lines 9-19**

FIG. 4 illustrates the procedure 400 used by a sender 102 (FIG. 1) in country X who wishes to send encrypted messages to a receiver 104 in country Y using an independently specified session key. The inputs to the procedure 400 are (1) a secret key; (2) an application-specific portion of the recovery information; and (3) an optional secret random salt. **A salt is a random value used to increase the randomness of a plaintext. A salt is used only once.** If a secret random salt is provided to the procedure, then it will be called SALT0. Otherwise, **SALT0 is derived pseudorandomly from the specified secret key, as described below"**

In the above cited reference Johnson does not disclose or teach about a "**method for operating a digital information processing system that encrypts information from a plurality of transmitting devices to a receiving device or vice versa; the method comprising processor executed means for using the information contained in the random number to identify single or plurality of unique mathematical or logical functions identical at the both transmitting and the receiving devices.**" As stated in the cited reference the sender 102 uses the **generated salt values, together with the public keys of the key recovery** agents 108-114 and other information, to generate encrypted P and Q values. In my disclosure, a salt (random number) is **NOT** used the in conjunction with public keys. It is a well known and documented fact that to help reduce the risk of dictionary attacks, random bytes (so-called "salt") are appended to the original plain text before generating hashes. As it is also clear from the above discussion that salt is a random number that is used to increase the randomness of a **plaintext**. In my invention, random numbers are never directly used or mixed to increase the randomness of plaintext (data segments). The pre-negotiated bits' positions mutually agreed upon within a random number are merely used to identify the functions used to encrypt/decrypt data segments. In addition, a random number is not derived pseudorandomly from the specified secret key as per the teaching of Johnson above.

21.2. Claim Objection:
The following part of the Claim # 21 was rejected based on the citation reference quoted from Johnson **(col.14 lines 42-55)**

"means for using the information contained in the random number to identify single or plurality of unique mathematical or logical functions identical at the both transmitting and the receiving devices (col.14 lines 42-55)"

21.2A Response to the cited Objection
In **(col.14 lines 42-55)** Johnson teaches:

Col. 14 lines 42-55
The sender's and receiver's first and second key recovery agent key headers 914, 918, 924, 928 contain information on the public keys belonging to the key recovery agents. Referring to FIG. 10, each key header contains an encryption algorithm ID 946 (specifying the public key algorithm to be used), the key length 948 in bits of the public key, and a key ID 950. The key ID 950 permits the receiver to determine the public keys under which the P and Q values are encrypted. The receiver needs these public keys in order to validate the encrypted P and Q values 602 and 604. The key IDs 950 permit the key recovery agents to determine the public keys under which the P and Q values are encrypted, and hence the private keys needed to decrypt the P and Q values.

The above cited reference does not disclose **"using the information contained in the random number to identify single or plurality of unique mathematical or logical functions identical at both the transmitting and receiving devices."** In my invention, there are **NO** key headers that contain the encryption algorithm IDs to specify the public key algorithm to be used. The above citation presents no reference of calculating the numeric values of the specific bits' locations within a random number mutually agreed upon in advance to determine the functions that will be used for data encryption. Johnson talks about recovering the session key through the use of a public key. It also refers to the public key under which P and Q values are encrypted. It also has a reference that each key header contains an encryption algorithm ID 946. In my invention there are no key headers to contain encryption algorithms.

21.4 Claim Objection:
The following part of the Claim # 21 was rejected based on the citation reference quoted from Johnson **(col.12 lines 19-25)**

"means for encrypting any type of digital information consisted of any arbitrary length segment through operating the mathematical or logical functions (col.12 lines 19 - 25);"

21.4A Response to the cited Objection
In **col.12 lines 19-25)** Johnson teaches:

Col.12 lines 19-25
Referring to FIG. 4, the sender 102 and receiver 104 first establish a secret session key K (step 402). The sender 102 and receiver 104 may

use whatever key distribution or key agreement procedure that they desire for establishing the key K. Typically the session key K will be a symmetric encryption key that is used both by the sender 102 to encrypt messages for the receiver 104 and by the receiver to decrypt messages from the sender".

The above reference talks about encrypting/decrypting messages through the use of session keys which are symmetric. Encrypting/Decrypting a message through information strictly contained within a symmetric key is completely different from selecting a range of mathematically and logical functions based on the numeric values of pre-negotiated bits' locations found within a random number. In addition, the related claim has been amended to clearly define this difference.

21.5 Claim Objection:
The following part of the Claim # 21 was rejected based on the citation reference quoted from Johnson **(col.16 lines 46-58; col. 17 lines 47-50)**

"means for encrypting the seed random number through operating the, mathematical or logical functions and declaring the resulting number as the seed random number for the next round (col. 16 lines 46-58; col. 17 lines 47-50);

21.5A Response to the cited Objection
In **col.16 lines 46-58)** Johnson teaches:

Col. 16 lines 46-58
The salts (SALT<for Px>, SALT<for Qx>, SALT<for Py>, and SALT<for Qy>) **protect the encrypted P and Q values** 602-608. The salt values (SALT<for Px>, SALT<for Qx>, SALT<for Py>, and SALT<for Qy>) are specifically constructed to be different. **In the case where SALT0 is a secret random value specified as an input to the encryption procedure**, this guarantees that every block to be encrypted for a key recovery agent has a dependency on SALT0 (a secret random 160-bit value independent of the key). In the case where SALT0 is pseudorandomly generated from the key, this guarantees that every block to be encrypted for a key recovery agent has a dependency on the entire key in a pseudorandom way.

The term "salt" is explained by Johnson in **col. 12 lines 9-19**

FIG. 4 illustrates the procedure 400 used by a sender 102 (FIG. 1) in country X who wishes to send encrypted messages to a receiver 104 in country Y using an independently specified session key. The inputs to the procedure 400 are (1) a secret key; (2) an application-specific portion of the recovery information; and (3) an optional secret random salt. **A salt is a random value used to increase the randomness of a plaintext. A salt is used only once.** If a secret random salt is provided to the procedure, then it will be called SALT0. Otherwise, SALT0 is derived pseudorandomly from the specified secret key, as described below"

In the above cited reference Johnson does not disclose or teach about "**encrypting the seed random**

number through operating the mathematical or logical functions and declaring the resulting number as the seed random number for the next round." As mentioned in the previous responses, it is a well known and documented fact that to help reduce the risk of dictionary attacks, random bytes (so-called "salt") are appended to the original plain text before generating hashes. As clearly stated in Johnson's above cited reference that salt is a "secret random value." In my invention, a random number is NOT considered secret and its contents are sent over an unsecured communicational channel. As it is clear from the above discussion that salt is a random number used to increase the randomness of a plaintext. In my invention a random number is never directly used or mixed to increase the randomness of plaintext (data segments). The pre-negotiated bits' positions found within a random number are merely used to identify the functions used to encrypt/decrypt data segments. In addition, a random number is not derived pseudorandomly from the specified secret key as per teaching of Johnson above.

21.5B Response to the cited Objection

In **col.17 lines 47-50)** Johnson teaches:

Col. 17 lines 47-50

"If there are "i" such blocks, then "i" different salt values are calculated, thereby ensuring that a different (but predictable) salt value is calculated for each of the m-bit blocks to be encrypted"

In order to fully understand the meaning and coverage of the above paragraph we need to read the entire context from **col. 17 lines 42-50**

When the length of P or Q is >m (where m is the maximum length of P or Q that can be encrypted with the public key of the intended key recovery agent), the value (P or Q) is divided into blocks of m bits. The last block may be a short block. If there are "i" such blocks, then "i" different salt values are calculated, thereby ensuring that a different (but predictable) salt value is calculated for each of the m-bit blocks to be encrypted. For example, suppose that m=256 and the length of P is 512 bits. In that case, P is split into two blocks of size 256 bits and two salt values are calculated using the above described algorithm

Again in the above cited reference Johnson does not disclose or teach about **"encrypting the seed random number through operating the mathematical or logical functions and declaring the resulting number as the seed random number for the next round."** Johnson talks about dividing P or Q into blocks when the length of P or Q is larger than 'm'. In my invention, there is no requirement to divide a data segment into single or multiple blocks. A data segment can consist of an arbitrary length as long as the system resources utilized at the sending and receiving devices can process it. Johnson also talks about correlation of "i" such blocks with "i" different salt values (random numbers). The methods discussed in my invention do not depend or need any of these defined correlations. In other words, there is no correlation or dependency between "i" salt values (random number) and "i" number of blocks in my invention.

21.6. Claim Objection:

The following part of the Claim # 21 was rejected based on the citation reference quoted from Johnson **(col.17 lines 1-11)**

"means for identifying the number of encryption rounds,. N, through the use of any information means mutually agreed between the transmitting and the receiving devices (col.17 lines 1-11)"

21.6A Response to the cited Objection

In **col.17 lines 1-11** Johnson teaches:

Col. 17, lines 1-11
Referring to FIG. 11, H(T1), the P or Q value, the indicator, and the salt are formatted in a block (1102), **padded as necessary on the left with 0 bits**, and **encrypted with the public key** of the key recovery agent, preferably using the enhanced optimal asymmetric encryption (EOAE) procedure described in the above-identified article of D. B. Johnson et al. As described in that article, the EOAE procedure contemplates first subjecting the formatted block 1102 to a plurality of masking rounds 1104 (in which one input half is alternating used to mask the other input half) before encrypting (1106) the result of the masking rounds.

In the above cited reference Johnson does not teach or suggest about **"identifying the number of encryption rounds, N, through the use of any information means mutually agreed upon between the transmitting and the receiving devices."** In the above reference there is no indication about mutually agreeing upon encryption/decryption rounds being performed on data segments. It talks about "masking rounds" in reference to recovering session keys which are completely different from actually performing variable rounds of encryption/decryption rounds on data segments. As defined by Johnson "a masking round" is in which one input half is alternating used to mask the other input half. In my invention, a variable number of mathematical or logical functions directly performs encryption rounds without the need of masking rounds. In my disclosure the **number 'N' which indicates the total number of rounds used for encryption/decryption process** is variable and not fixed. This number is dynamically determined through the information contained in a random number or through the encrypted version of the random number.

21.7 Claim Objection:
The following part of the Claim # 21 was rejected based on the citation reference quoted from Johnson **(col.16 lines 46-58; col. 17 lines 1-11;col. 17 lines 47-50)**

"means for repeating the encryption process on the said digital information segment and the said. random number for N number of rounds (col.16 lines 46-58; col.17 lines 1-11; col.17 lines 47-50);"

21.7A Response to the cited Objection
In **col.16 lines 46-58** Johnson teaches:

Col. 16 lines 46-58
The salts (SALT<for Px>, SALT<for Qx>, SALT<for Py>, and SALT<for Qy>) protect the encrypted P and Q values 602-608. The salt values (SALT<for Px>, SALT<for Qx>, SALT<for Py>, and SALT<for Qy>) are specifically constructed to be different. In the case where SALT0 is a secret random value specified as an input to the encryption procedure, **this guarantees that every block to be encrypted for a key recovery agent has a dependency on SALT0 (a secret random 160-bit value independent of the key).** In the case where SALT0 is pseudorandomly generated from the key, this guarantees that every block to be encrypted for a key recovery agent has a dependency on the entire key in a pseudorandom way.

The term "salt" is explained by Johnson in **col. 12 lines 9-19**

FIG. 4 illustrates the procedure 400 used by a sender 102 (FIG. 1) in country X who wishes to send encrypted messages to a receiver 104 in country Y using an independently specified session key. The inputs to the procedure 400 are (1) a secret key; (2) an application-specific portion of the recovery information; and (3) an optional secret random salt. **A salt is a random value used to increase the randomness of a plaintext. A salt is used only once.** If a secret random salt is provided to the procedure, then it will be called SALT0. Otherwise, SALT0 is derived pseudorandomly from the specified secret key, as described below"

In the above cited reference Johnson does not disclose or teach about "**repeating the encryption process on the said digital information segment and the said random number for N number of rounds.**" As clearly stated in Johnson's cited reference, salt is a "secret random value." In my invention, a random number is **NOT** considered secret and its contents are sent over an unsecured communication channel. In addition, it specifically states that "every block to be encrypted.....has a dependency on SALT0." In my invention, a data segment that is encrypted **does not** have a direct dependency on a random number, e.g., SALT0. As it is clear from the above discussion that salt is a random number used to increase the randomness of a **plaintext**. In my invention a random number is never directly used or mixed to increase the randomness of plaintext (data segments). The pre-negotiated bits' positions found within a random number are merely used to identify the functions used to encrypt/decrypt data segments. In addition, a random number is not derived pseudorandomly from the specified secret key as per the teaching of Johnson above. The other references have already been discussed in the previous responses on this particular reference. In addition, the related claim has been amended to clearly define this difference.

Responses to the individual Objections as cited in reference to Claim No. 22

22.1 Claim Objection:
The following part of the Claim # 22 was rejected based on the citation reference quoted from Johnson **(col.17 lines 1-11)**

"Regarding claim 22, Johnson teaches a method for operating a digital information, processing system that decrypts information from a plurality of transmitting. devices to a receiving device or vice versa the method comprising processor executed steps of:

at the receiving device:

means for receiving and identifying the seed random number of an arbitrary length from the transmitting device;

means for identifying the number of encryption rounds, N, through any means mutually agreed between the transmitting and the receiving devices (col.17 lines .1-11);"

22.1A Response to the cited Objection
In **col.17 lines 1-11** Johnson teaches:

Col. 17, lines 1-11
Referring to FIG. 11, H(T1), the P or Q value, the indicator, **and the salt** are formatted in a block (1102), padded as necessary on the left

with 0 bits, and encrypted with the public key of the key recovery agent, preferably using the enhanced optimal asymmetric encryption (EOAE) procedure described in the above-identified article of D. B. Johnson et al. As described in that article, the EOAE procedure contemplates first subjecting the formatted block 1102 to a plurality of masking rounds 1104 (in which one input half is alternating used to mask the other input half) before encrypting (1106) the result of the masking rounds.

Johnson in the above reference does not teach or discuss about **"receiving and identifying the seed random number of an arbitrary length from the transmitting device and identifying the number of encryption rounds, N, through any means mutually agreed upon between the transmitting and the receiving devices."**
In the above description there is no indication about repeating the encryption/decryption rounds in accordance with the disclosed techniques described in my invention. The above reference talks about "formatting a block constituting "H(T1), the P or Q value the indicator and the salt." This quotation confirms a drastic difference in which a salt (random number) is used in Johnson's disclosure versus my invention. In my invention, a random number (salt) is **never** mixed or appended with plaintext. As mentioned before, it talks about plurality of masking rounds in reference to extracting session keys. As further defined by Johnson, a masking round constitutes "one input half is alternating used to mask the other input half." In my invention, plurality of mathematical or logical functions operate as a round of encryption on the data segments and not on the session keys.

22.2 Claim Objection:
The following part of the Claim # 22 was rejected based on the citation reference quoted from Johnson **(col.14 lines 45-47)**

"means for using the information contained within the specific bits of the seed random number to identify a single or plurality of unique mathematical or logical functions (col.14 lines 45-47);"

22.2A Response to the cited Objection
In **col.14 lines 45-47** Johnson teaches:

Col.14 lines 45-47
The sender's and receiver's first and second key recovery agent key headers 914, 918, 924, 928 contain information on the public keys belonging to the key recovery agents. Referring to FIG. 10, each key header contains an encryption algorithm ID 946 (specifying the public key algorithm to be used), the key length 948 in bits of the public key, and a key ID 950

The above reference does not disclose about **"using the information contained within the specific bits of the seed random number to identify a single or plurality of unique mathematical or logical functions."** As clearly stated, "key headers" contain the information about the "public key." In my disclosure there are **NO** "key headers" to contain information about private/public key infrastructure. In addition, there are **NO** key headers that contain or carry encryption algorithm IDs. As stated earlier in my invention, pre-negotiated bits within a random number are used to determine the order of mathematical functions which operate on data segments to produce encryption.

22.3 Claim Objection:
The following part of the Claim # 22 was rejected based on the citation reference quoted from Johnson

(col. 8 lines 11-40)

"means for identifying single or plurality of inverse functions corresponding to each of the identified mathematical or logical functions (col.8 lines 11-40);"

22.3A Response to the cited Objection
In **col. 8 lines 11-40** Johnson teaches:

"Key inversion function 204 is invertible in the sense that: (1) it provides a one-to-one mapping between the key K and the generated key recovery values; and (2) the key K can easily be regenerated from the generated key recovery values by inverting the function. If the generated key recovery values include only P and Q (i.e., R is not generated), then the key K is completely determined by P and Q and can be trivially regenerated given knowledge of these values. If the generated key recovery values also include R, then the key is completely determined only by P, Q and R, and cannot be trivially determined from only P and Q. However, the number of possible R values generated by key inversion function 204 is made low enough that a law enforcement agent knowing P and Q can feasibly regenerate the key K by exhausting the space of possible R values, as described below. The work factor required to ascertain R is intended to discourage routine decryption of messages by law enforcement, even if it obtains the P and Q values.

To decipher a message that it has intercepted, a law enforcement agent 116 extracts the **encrypted key recovery values P' and Q' from the session header** and presents them to key recovery agents 108, 110, together with evidence of proper authority (such as a court order). Upon satisfying themselves of the law enforcement agent's authority, **the key escrow agents 108, 110 decipher the encrypted key recovery values using their private decryption keys** and give the recovered values P and Q to the law enforcement agent 116. The law enforcement agent then generates successive trial values of R and supplies them together with the recovered P and Q values as inputs to the key inversion function 204 until the original session key K is recovered."

The above cited reference does not disclose a reference to **"identifying single or plurality of inverse functions corresponding to each of the identified mathematical or logical functions."** The context of my cited claim refers to the encryption operation on a data segment performed first by a plurality of mathematical or logical functions in a known order at the sending side and then at the receiving side by their corresponding inverse functions in the known order to decrypt the data segment. Johnson teaches about extracting "encrypted key recovery values P' and Q' from the session header. In my disclosure, there are no recovery values 'P' and 'Q' contained in a session header. In addition, the related claim is amended to truly reflect the scope of the invention.

22.4. Claim Objection:
The following part of the Claim # 22 was rejected based on the citation reference quoted from Johnson **(col.16 lines 46-58 col. 17 lines 47-50)**

"means for encrypting the seed random number through operating the mathematical or logical

functions and declaring the resulting number as the seed random number for the next round (col.16 lines 46-58; col. 17 lines 47-50);"

22.4A Response to the cited Objection
In **col. 16 lines 46-58** Johnson teaches:

Col. 16 lines 46-58
The salts (SALT<for Px>, SALT<for Qx>, SALT<for Py>, and SALT<for Qy>) **protect the encrypted P and Q** values 602-608. The salt values (SALT<for Px>, SALT<for Qx>, SALT<for Py>, and SALT<for Qy>) are specifically constructed to be different. **In the case where SALT0 is a secret random value specified as an input to the encryption procedure**, this guarantees that every block to be encrypted for a key recovery agent has a dependency on SALT0 (a secret random 160-bit value independent of the key). In the case where SALT0 is pseudorandomly generated from the key, this guarantees that every block to be encrypted for a key recovery agent has a dependency on the entire key in a pseudorandom way.

The term "salt" is explained by Johnson in **col. 12 lines 9-19**

FIG. 4 illustrates the procedure 400 used by a sender 102 (FIG. 1) in country X who wishes to send encrypted messages to a receiver 104 in country Y using an independently specified session key. The inputs to the procedure 400 are (1) a secret key; (2) an application-specific portion of the recovery information; and (3) an optional secret random salt. **A salt is a random value used to increase the randomness of a plaintext. A salt is used only once.** If a secret random salt is provided to the procedure, then it will be called SALT0. Otherwise, SALT0 is derived pseudorandomly from the specified secret key, as described below"

In the above cited reference Johnson does not disclose or teach about "**encrypting the seed random number through operating the mathematical or logical functions and declaring the resulting number as the seed random number for the next round.**" As stated above that the primary function of "salts" (random numbers) in Johnson's invention is to "protect the encrypted 'P' and 'Q' values." The random number is not encrypted by plurality of mathematical or logical functions as it is done in my invention. As stated before 'salts' are random bytes that are appended with the plain text before generating hashes. As clearly stated in Johnson's cited reference that salt is a "secret random value." In my invention, a random number is **NOT** considered secret and its contents are sent over an unsecured communication channel. Johnson contends to use 'salt' in order to increase the randomness of a **plaintext**. In my invention a random number is never used to mix it up with the plaintext (in this case, data segments) to increase their randomness. The pre-negotiated bits' positions found within a random number are merely used to identify the functions used to encrypt/decrypt data segments. In addition, a random number is not derived pseudorandomly from the specified secret key as per the teaching of Johnson above.

22.4B Response to the cited Objection
In **col. 17 lines 47-50** Johnson teaches:

Col. 17 lines 47-50

If there are "i" such blocks, then "i" different salt values are calculated, thereby ensuring that a different (but predictable) salt value is calculated for each of the m-bit blocks to be encrypted

22.5 Claim Objection:
The following part of the Claim # 22 was rejected based on the citation reference quoted from Johnson **(col. 8 lines 37-40; col. 16 lines 46-58; col. 17 lines 1-11; col. 17 lines 47-50)**

"means for repeating the decryption process on the received digital information segment for N number of rounds to remove any effects of encryption on the said digital information segment (col. 8 lines 37-40; col. 16 lines 46-58; col. 17 lines 1-11; col. 17 lines 47-50)"

22.5A. Response to the cited Objection
In **col. 8 lines 37-40** Johnson teaches:

"The law enforcement agent then generates successive trial values of R and supplies them together with the recovered P and Q values as inputs to the key inversion function 204 until the original session key K is recovered."

In the above description there is no reference about **repeating the decryption process on the received digital information segment for N number of rounds to remove any effects of encryption on the said digital information segment.** The above description refers to the successive "trial values" of R together with recovered P and Q values as input to the key inversion function until the original session key is recovered." As clearly stated, an agent generates the "successive trial values" which are not exactly the known functions for decrypting data segments as disclosed in accordance with decrypting methods presented in my invention.

22.5B Response to the cited Objection
In the cited reference **col. 16 lines 46-58** as presented with 22.4A response there is no disclosure of **repeating the decryption process on the received digital information segment for N number of rounds to remove any effects of encryption on the said digital information segment.** As clearly stated in Johnson's cited reference, salt is a "secret random value." In my invention, a random number is **NOT** considered secret and its contents are sent over an unsecured communication channel. As it is clear from the above discussion that salt is a random number to increase the randomness of a **plaintext**. In my invention a random number is never directly used or mixed to increase the randomness of plaintext (data segments). The pre-negotiated bits' positions found within a random number are merely used to identify the functions used to encrypt/decrypt data segments. In addition, a random number is not derived pseudorandomly from the specified secret key as per teaching of Johnson above.

22.5C Response to the cited Objection
In **col. 17 lines 47-50** Johnson teaches:

"If there are "i" such blocks, then "i" different salt values are calculated, thereby ensuring that a different (but predictable) salt value is calculated for each of the m-bit blocks to be encrypted."

In order to fully understand the meaning and coverage of the above paragraph we need to read the entire context from **col. 17 lines 42-50**

"When the length of P or Q is >m (where m is the maximum length of P

or Q that can be encrypted with the public key of the intended key recovery agent), the value (P or Q) is divided into blocks of m bits. The last block may be a short block. If there are "i" such blocks, then "i" different salt values are calculated, thereby ensuring that a different (but predictable) salt value is calculated for each of the m-bit blocks to be encrypted. For example, suppose that m=256 and the length of P is 512 bits. In that case, P is split into two blocks of size 256 bits and two salt values are calculated using the above described algorithm"

Again in the above cited reference Johnson does not disclose or teach about "**repeating the decryption process on the received digital information segment for N number of rounds to remove any effects of encryption on the said digital information segment.**" Johnson first talks about dividing P or Q into blocks of 'm' bits (if the length of P or Q is larger than 'm'). In my invention, there is no requirement to divide a data segment into single or multiple blocks. A data segment can consist of an arbitrary length as long as the system resources utilized at the sending and receiving devices can process it. Furthermore, according to Johnson's teaching, if there are 'i' blocks which result from the division then he recommends using 'i' different "salt values" (random number); one salt value for each block. It must be noted that each 'i' salt value is appended with each 'i' resulting block. The methods discussed in my invention do not depend or need any of these defined correlations. In other words, there is no correlation or dependency between "i" salt values (random number) and "i" number of blocks in my invention. In addition, the related claim has been amended to precisely address this particular issue and subject matter.

Conclusion

As it is clear from the above discussion, Johnson's subject matter and area of discussion is completely different than the Applicant's. Johnson's invention is directed toward recovering session keys typically used in symmetric, e.g., DES or asymmetric public/private key infrastructure. It does not present or discuss any encryption techniques that can actually encrypt data. The Applicant's invention does not present any techniques for recovering session keys. The Applicant's disclosure is strictly about presenting an encryption/decryption technique that actually encrypt/decrypt data segments.

In addition, the techniques deployed to recover session keys by Johnson are not similar or analogous to the Applicant's methods and techniques.

Applicant submits the claims that are in condition of allowance, and notice of such is respectfully requested. If after review, the Examiner feels that there are further unresolved issues, the Examiner is invited to call the Applicant at 949-457-1243.

Respectfully submitted,

By Shakeel Mustafa

Address: 24831 Hendon St
Laguna Hills, CA 92653

Tel: 949-457-1243
949-510-6023